d. Repeating steps a-c until modes defined at any level in the hierarchy are mapped to sub-modes down the hierarchy, whereby when said hierarchy is executing, only active components will execute throughout the hierarchy.

52. (New) A method as recited in claim 51 wherein said hierarchical system is a control system.

53. (New) A method as recited in claim 51 wherein said hierarchical system is a real-time control system.

54. (New) A method as recited in claim 51 wherein said components are represented in a graphical user interface and said choosing said subset of components includes selecting said sub-modes using said graphical user interface.

55. (New) A method as recited in claim 51 wherein said components are represented in a graphical user interface and said choosing said submodes includes selecting said subset of components using said graphical user interface, whereby said subset of components are visually distinguished from components not in said subset.

56. (New) A method as recited in claim 51 wherein said hierarchical system is a control system, and said components are executing said control system on a computer.

57. (New) A method as recited in claim 56 further comprising:
switching from said mode to a different mode while said control system is executing;
deactivating those components that correspond to said mode; and
activating those components that correspond to said different mode.

58. (New) A method of mapping a thread of a processor to components within a control system, wherein said control system comprises a plurality of components in a multi-level hierarchy, said method comprising:

a. defining one or more logical rates of execution at each level in said multi-level hierarchy;

b. for each component in a level of said multi-level hierarchy, assigning said component to one or more of said logical rates; and

c. mapping said thread of said processor to one or more of said logical rates, thus mapping said thread through said multi-level hierarchy, resulting in a system of components, executing at the actual execution rate of said thread.

59. (New) A method as recited in claim 58 wherein said control system is a real-time control system.

60. (New) A method as recited in claim 58 wherein said components are represented in a graphical user interface and said process of assigning includes selecting components at each level using said graphical user interface, and defining said logical rates of execution using said graphical user interface..

61. (New) A method as recited in claim 58 wherein said components are executing on a real-time computer.

62. (New) A method as recited in claim 58 wherein multiple threads are mapped to multiple components, wherein each component may be assigned one or multiple logical rates and each said logical rate mapped to a thread.

63. (New) A method of defining an executable image for a control system comprising a plurality of components, said method comprising

a. choosing a subset of said components to correspond to said executable image;

b. identifying a name for said executable image and its corresponding subset of said components; and

c. for each chosen component in said subset, indicating that said component is part of said executable image for said control system, whereby when said executable image is produced for said control system, only said subset of said components that correspond to said executable image will be included.

64. (New) A method as recited in claim 63 wherein said control system is an electromechanical control system.

65. (New) A method as recited in claim 63 wherein said control system is a real-time control system.

66. (New) A method as recited in claim 63 wherein said received components are represented in a graphical user interface and said element of choosing includes selecting said subset of components using said graphical user interface, whereby said subset of components are visually distinguished from components not in said executable image.

67. (New) A method as recited in claim 63 further comprising:

a. loading components corresponding to said executable image into a real-time computer; and

b. executing said executable image on said real-time computer.

68. (New) A method as recited in claim 63 whereby multiple executable images are generated, each destined for a different particular computer.

69. (New) A method of defining an executable image for a control system, wherein said control system comprises a multi-level hierarchy, said method further comprising:

a. defining at least one logical executable name or designator at each level;

b. choosing a subset of components at each level to correspond to each said logical executable name;

c. for each chosen component in each subset, indicating that said component is part of said logical executable name;

d. continuing this process throughout the hierarchy, thus assigning components to logical executable names at each level; and

e. mapping said executable image to said logical executable names throughout said multi-level hierarchy such that only the subset of said components assigned to the mapped said logical executable names will be included in the executable image.

70. (New) A method as recited in claim 69 wherein said control system is a real-time control system.

71. (New) A method as recited in claim 69 wherein said components are represented in a graphical user interface and said element of choosing includes selecting said subset of components using said graphical user interface, whereby said subset of components are visually distinguished from components not in said executable image.

72. (New) A method as recited in claim 69 further comprising:

a. loading components corresponding to said executable image into a real-time computer; and

b. executing said executable image on said real-time computer.

73. (New) A method as recited in claim 69 whereby multiple executable images are generated, each destined for a different particular computer.

REMARKS

Claims 1-50 have been cancelled. Claims 51-73 have been added.

The applicants would like to thank the Examiner for his time during the teleconference of July 15, 2003.

The Examiner rejected claims 23-30 and 45 under 35 U.S.C. 103(a) as being unpatentable over the commercial product Rational Rose 4.0 released in 1996 in view of Steinman et al. (USPN 6,259,958 hereinafter Steinman). The Examiner rejected claims 31-44 under 35 U.S.C. 103(a) as

being unpatentable over the commercial product Rational Rose 4.0 released in 1996 in view of Steinman and further in view of Gretta. Independent claims 23, 31, 37, 38, and 45 have been amended to more clearly distinguish the recitation of a control system with components from an object oriented programming language, as disclosed in Rational Rose. For at least these reasons, the claims are not obvious over Rational Rose 4.0 in view of Steinman and further in view of Gretta.

New claims 51-73 have been added. New claim 51 is related to cancelled claim 23. Claim 51 adds limitations related to a multi-level hierarchy that was not in claim 23.

Respectfully submitted,
BEYER WEAVER & THOMAS, LLP

Michael Lee
Reg. No. 31,846

P.O. Box 778
Berkeley, CA 94704-0778
(831) 655-2300

TRANSMISSION VERIFICATION REPORT

TIME : 07/18/2003 20:33

DATE,TIME	07/18 20:33
FAX NO./NAME	17037467240
DURATION	00:00:00
PAGE(S)	00
RESULT	BUSY
MODE	STANDARD

BUSY: BUSY/NO RESPONSE